

AZOR Ophélie
05/05/2024

PRÉSENTATION

Menu Maker by
Qwenta



Sommaire

1. *Contexte du projet*
2. *Aperçu de la maquette*
3. *Méthodologie utilisée*
4. *Tableau Kanban*
5. *Spécifications techniques*
6. *Veille technologique*
7. *Conclusion*
8. *Questions*

Contexte du Projet

- **Qwenta**, une entreprise dédiée à fournir des solutions numériques innovantes pour les restaurateurs.
- **Menu Maker** est une plateforme en ligne conçue pour aider les restaurateurs à gérer et personnaliser leurs menus de manière intuitive et professionnelle. L'objectif est de simplifier la création, la gestion et la diffusion des menus tout en offrant une expérience utilisateur fluide et personnalisée.

Objectifs Principaux :

1. Création et Personnalisation des Menus :

- Concevoir des menus attrayants avec ajout de plats, personnalisation du style (typographies, couleurs, photos).
- Organiser les offres culinaires par catégories et éléments personnalisés.

2. Diffusion et Exportation :

- Exporter les menus en PDF pour impression ou distribution numérique.
- Partager directement sur des plateformes tierces comme Instagram et Deliveroo.

3. Gestion Simplifiée :

- Tableau de bord pour gérer, suivre et mettre à jour les menus.
- Interface sécurisée avec gestion des informations utilisateur et options d'authentification

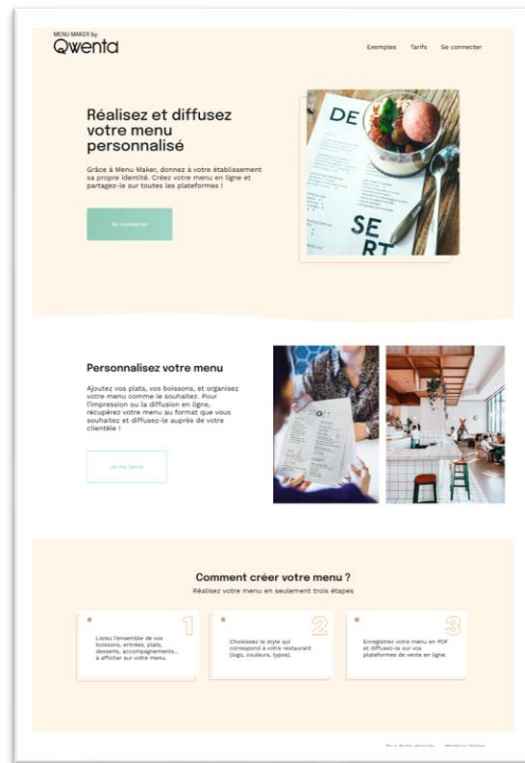
4. Accès et Support :

- Interface utilisateur claire avec accès à des informations et support intégré.
- Mentions légales accessibles pour garantir transparence et conformité.

Aperçu de la maquette

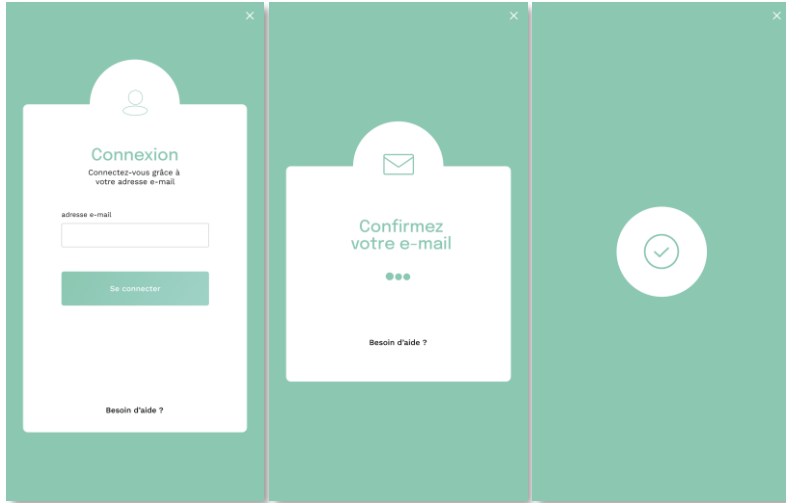
La maquette sur Figma est essentielle pour la réussite du projet Menu Maker. Elle assure une compréhension claire des attentes du client, optimise le processus de développement, et améliore la collaboration entre les équipes. En fournissant une représentation visuelle précise, elle facilite la création d'un produit final qui répond aux besoins et dépasse les attentes.

[Lien vers la maquette Figma](#)

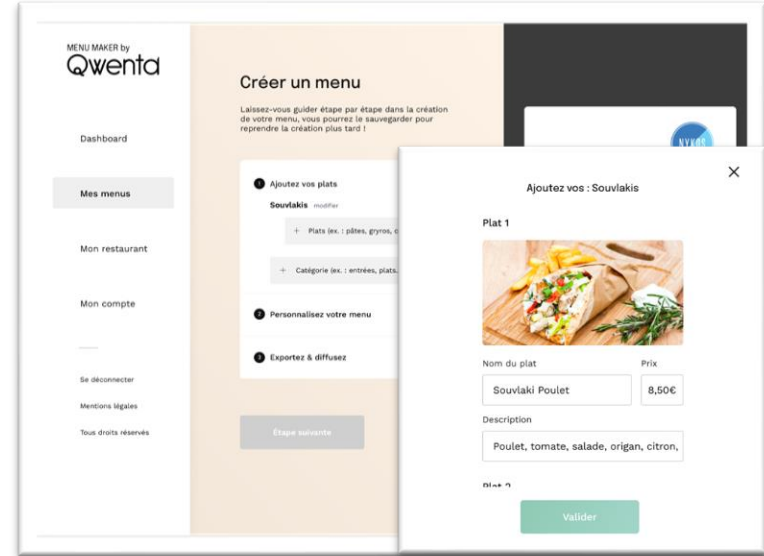


Aperçu de la maquette

- *Démonstration des fonctionnalités clés*



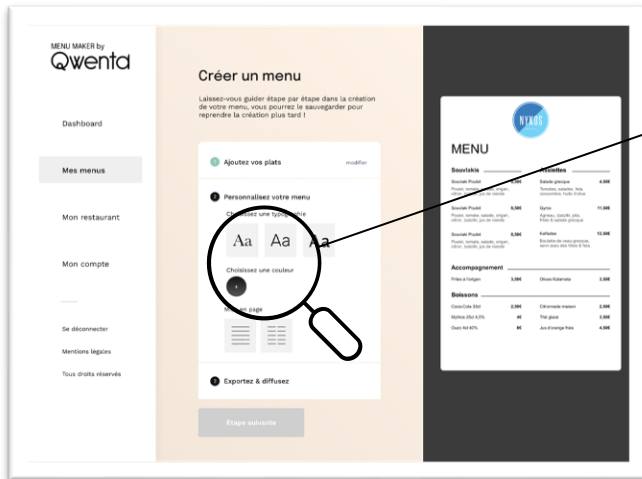
Connexion ou inscription au site avec une adresse e-mail unique qui va permettre recevoir un mail de confirmation



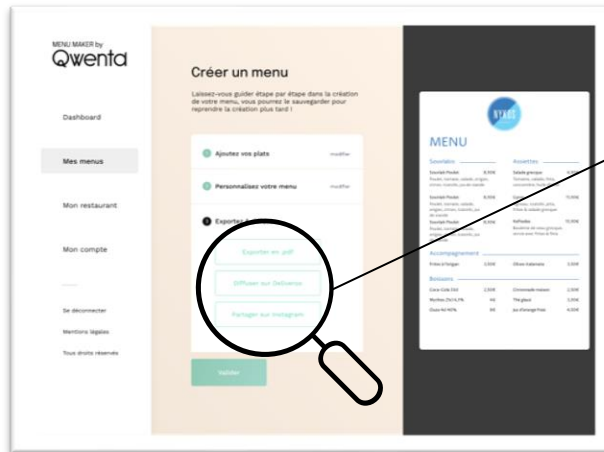
La création d'un menu simplifié qui va contenir des informations comme le nom, le prix et la description du plat

Aperçu de la maquette

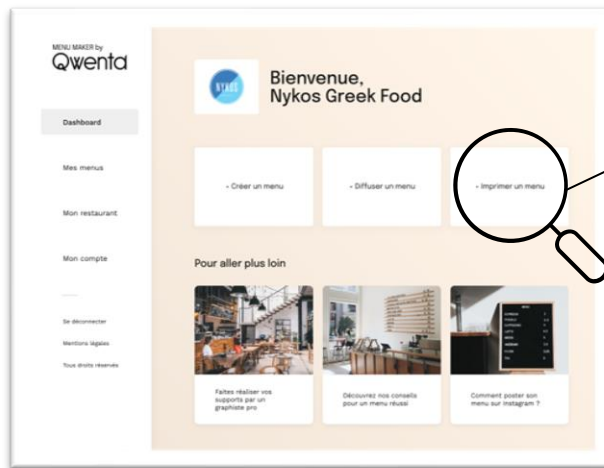
- *Démonstration des fonctionnalités clés*



La possibilité de personnaliser son menu en modifiant la typographie et les couleurs avant la mise en page



Lorsque le menu sera créé, il pourra être facilement exporté en PDF, partagé sur Instagram, et diffusé sur Deliveroo



Le restaurateur pourra commander l'impression d'un menu

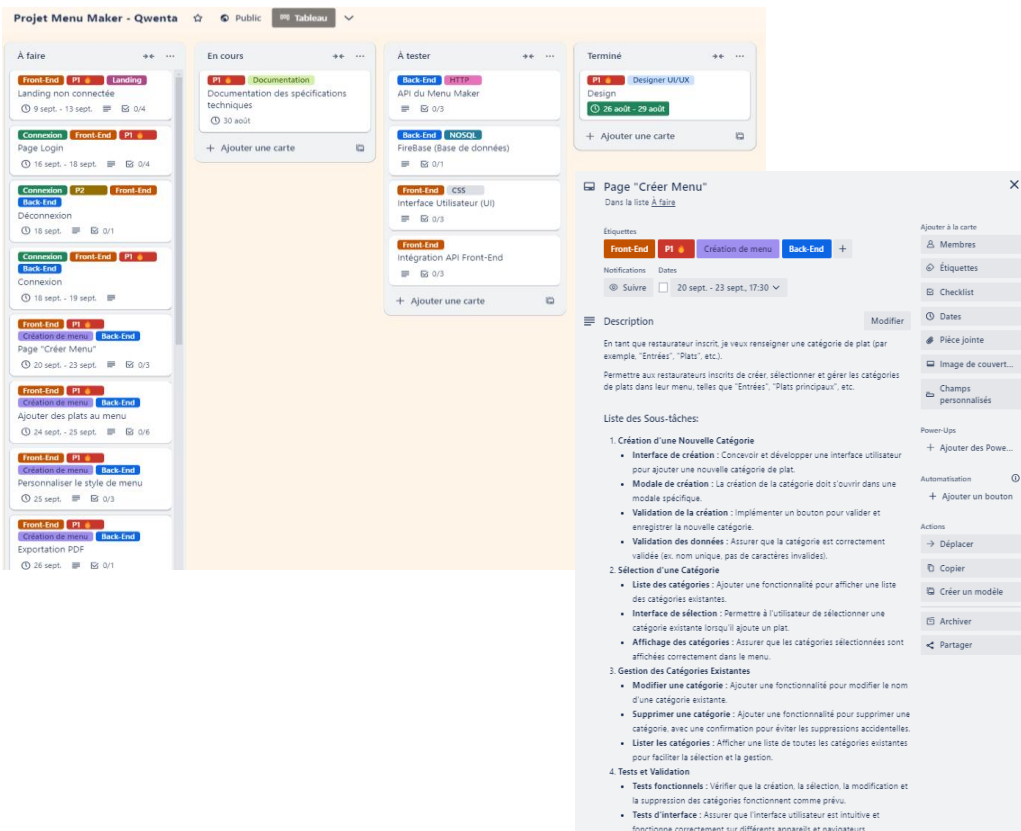
Méthodologie utilisée

- **Méthodologie Agile** : Agile est une approche flexible qui favorise les itérations rapides et l'adaptation continue aux besoins des utilisateurs. Elle se concentre sur la collaboration constante et les ajustements réguliers du projet en fonction des retours.
- **Scrum** : Scrum est un cadre Agile qui organise le développement en sprints de 1 à 4 semaines. Chaque sprint se termine par une revue et une rétrospective, permettant des ajustements rapides et une amélioration continue. Les sprints permettent de livrer régulièrement des parties fonctionnelles du produit.
- Définition des sprints: Ce sont des cycles de développement fixes, généralement de 2 à 4 semaines, durant lesquels une partie fonctionnelle du produit est développée, testée et revue.
- **Nous aurons besoin de 3 sprints pour finir le projet, le premier sprint aura une durée de 3 semaines et les deux autres seront 2 semaines chacun, ce qui nous fait un total de 49 jours**
- En utilisant la méthodologie Agile et Scrum, le projet Menu Maker peut bénéficier d'une approche itérative et adaptable, permettant de répondre rapidement aux besoins des utilisateurs tout en maintenant une collaboration efficace et une amélioration continue du produit.

Suivi du projet avec le Kanban

Le tableau Kanban offre une vue d'ensemble visuelle de toutes les tâches du projet, organisées par état (À faire, En cours, Terminé). Cela permet de voir facilement où chaque tâche en est dans le processus.

[Lien vers le Kanban](#)



Projet Menu Maker - Qwenta Public Tableau

À faire

- Front-End P1 Landing 9 sept. - 13 sept. 0/4
- Convention Front-End P1 Page Login 16 sept. - 18 sept. 0/4
- Convention P2 Back-End Déconnexion 18 sept. 0/1
- Convention Front-End P1 Connexion 18 sept. - 19 sept.
- Front-End P1 Création de menu Page "Créer Menu" 20 sept. - 23 sept. 0/3
- Front-End P1 Création de menu Ajouter des plats au menu 24 sept. - 25 sept. 0/6
- Front-End P1 Création de menu Personnaliser le style de menu 25 sept. 0/3
- Front-End P2 Exportation PDF 26 sept. 0/1

En cours

- P1 Documentation Documentation des spécifications techniques 30 août

À tester

- Back-End P1 API du Menu Maker 0/3
- Back-End NOSQL Firebase (Base de données) 0/1
- Front-End CSS Interface Utilisateur (UI) 0/3
- Front-End Intégration API Front-End 0/3

Terminé

- P1 Design Designer UI/UX 26 août - 29 août

Page "Créer Menu"

Dans la liste À faire

Étiquettes: Front-End P1 Création de menu Back-End

Notifications: Suivre 20 sept. - 23 sept., 17:30

Description

En tant que restaurateur inscrit, je veux renseigner une catégorie de plat (par exemple, "Entrées", "Plats", etc.).

Permettre aux restaurateurs inscrits de créer, sélectionner et gérer les catégories de plats dans leur menu, telles que "Entrées", "Plats principaux", etc.

Liste des Sous-tâches:

- Création d'une Nouvelle Catégorie**
 - Interface de création:** Concevoir et développer une interface utilisateur pour ajouter une nouvelle catégorie de plat.
 - Modale de création:** La création de la catégorie doit s'ouvrir dans une modale spécifique.
 - Validation de la création:** Implémenter un bouton pour valider et enregistrer la nouvelle catégorie.
 - Validation des données:** Assurer que la catégorie est correctement validée (ex: nom unique, pas de caractères invalides).
- Sélection d'une Catégorie**
 - Liste des catégories:** Ajouter une fonctionnalité pour afficher une liste des catégories existantes.
 - Interface de sélection:** Permettre à l'utilisateur de sélectionner une catégorie existante lorsqu'il ajoute un plat.
 - Affichage des catégories:** Assurer que les catégories sélectionnées sont affichées correctement dans le menu.
- Gestion des Catégories Existantes**
 - Modifier une catégorie:** Ajouter une fonctionnalité pour modifier le nom d'une catégorie existante.
 - Supprimer une catégorie:** Ajouter une fonctionnalité pour supprimer une catégorie, avec une confirmation pour éviter les suppressions accidentelles.
 - Lister les catégories:** Afficher une liste de toutes les catégories existantes pour faciliter la sélection et la gestion.
- Tests et Validation**
 - Tests fonctionnels:** Vérifier que la création, la sélection, la modification et la suppression des catégories fonctionnent comme prévu.
 - Tests d'interface:** Assurer que l'interface utilisateur est intuitive et fonctionne correctement sur différents appareils et navigateurs.

Outils Recommandés

- Front-End**
 - React.js:** Pour créer les composants interactifs comme les modales et les listes déroulantes.
 - Formik/Yup:** Pour la gestion des formulaires et la validation des données (ex: vérification de l'unicité des noms de catégorie).
 - Styled Components / Tailwind CSS:** Pour un design réactif et cohérent des interfaces.
- Back-End**
 - Node.js avec Express:** Pour gérer les requêtes de création, modification, et suppression des catégories.
 - Firestore:** Pour stocker les informations sur les catégories de plats.
- Tests et Validation**
 - Jest/Enzyme ou Cypress:** Pour effectuer des tests unitaires et des tests d'intégration sur les fonctionnalités front-end.
 - Lighthouse (Google Chrome):** Pour tester la performance, l'accessibilité, et la compatibilité de l'interface utilisateur.

Succès [Supprimer]

☐ Créer une nouvelle catégorie

☐ La création de catégorie s'ouvre dans une modale spécifique, et doit pouvoir être validée

☐ Sélectionner une catégorie créée précédemment

Ajouter un élément

Suivi du projet avec le Kanban

- User stories (US) et tâches attribuées :
- ✓ **À Faire** : Les tâches à réaliser seront placées dans cette catégorie du tableau, avec une User Story détaillée qui comprendra la description, les critères spécifiques à chaque tâche et les outils techniques nécessaires pour mener à bien ce projet.
- ✓ **En Cours** : Les tâches qui sont en cours de réalisation sont placées dans cette catégorie. Elles doivent être activement travaillées par un membre de l'équipe. Elles peuvent être en phase de développement, de design ou d'intégration
- ✓ **À Tester** : Une fois que les tâches sont complètes, elles sont déplacées dans la catégorie "À tester". Cette étape est importante pour vérifier que les tâches réalisées fonctionnent comme prévu et répondent aux critères d'acceptation. Les tâches doivent être prêtes pour les tests, incluant des vérifications manuelles ou automatisées pour s'assurer que les fonctionnalités sont conformes aux exigences.
- ✓ **Terminé** : Cette catégorie contient toutes les tâches qui ont été complètement réalisées, testées et acceptées. Elles sont considérées comme achevées et prêtes pour être intégrés au projet final. Les tâches doivent avoir passé les tests avec succès et être conformes aux critères d'acceptation définis dans la User Story.

Suivi du projet avec le Kanban

- Suivi et Coordination de l'équipe :
 - Cette disposition permet une *vue d'ensemble claire* et *immédiate* de l'état du projet. Les membres de l'équipe peuvent facilement suivre l'avancement des tâches, identifier les étapes du processus où des ralentissements pourraient se produire, et repérer les tâches nécessitant une attention supplémentaire pour rester dans les délais
 - Les *critères d'acceptation* définissent ce qui doit être accompli pour que la tâche soit considérée comme terminée, ce qui permet une évaluation précise des tâches. Les membres de l'équipe peuvent facilement voir les exigences spécifiques et s'assurer que chaque tâche répond aux normes avant de la déplacer vers la colonne "Terminé."
 - L'*attribution* des tâches clarifie qui est responsable de quoi, ce qui évite les chevauchements et les confusions. Cela aide également à équilibrer la charge de travail et assure que chaque tâche est prise en charge par la personne appropriée.

Trello facilite le suivi et la coordination de l'équipe grâce à sa vue visuelle intuitive, à la gestion claire des critères de réalisation, et à l'attribution précise des tâches. Ces fonctionnalités permettent une gestion de projet plus efficace en offrant une vue d'ensemble transparente, en assurant la conformité aux exigences, et en clarifiant les responsabilités au sein de l'équipe.

Spécifications techniques

1. Choix Technologiques

- Utilisation de *React* pour le front-end : modularité et dynamisme.
- *Node.js* avec *Express* pour le back-end : performances et scalabilité.
- Base de données NoSQL (MongoDb) : flexibilité dans la gestion des menus.

2. Liens avec les Back-End

- *API REST* pour la communication entre le front-end et le back-end
- Intégration de l'*API instagram* pour le partage des menus
- *API Deliveroo* pour la commande

3. Domaine et Hébergement:

- Domaine sous Qwenta
- Hébergement sur Google Cloud Patform

4. Accessibilité:

- Compatibilité avec les derniers navigateurs (Crome, Firefox, Safari)
- Visualisation sur un ordinateur, sans version mobile prévue pour l'instant

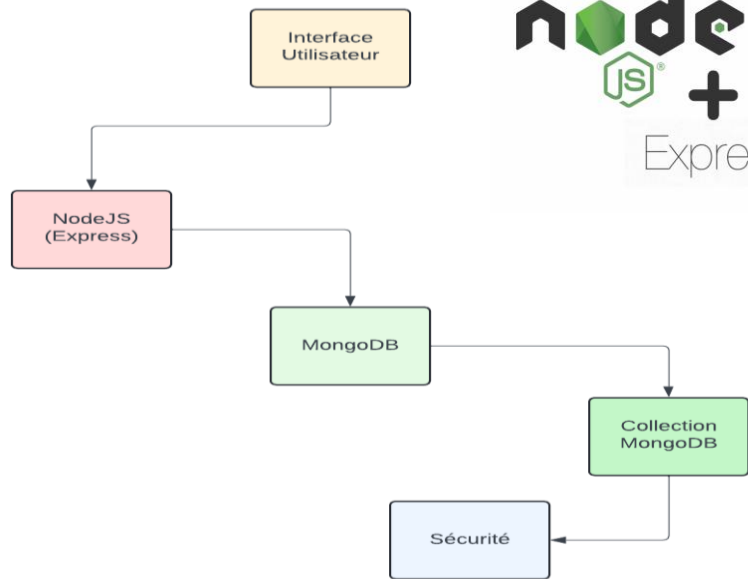
5. Recommandations en sécurité:

- JWT pour l'authentification sécurisée
- HTTPS pour chiffrer les communications
- Journalisation des accès et stockage sécurisé des mots de passe

6. Maintenance et Mise à jour futures:

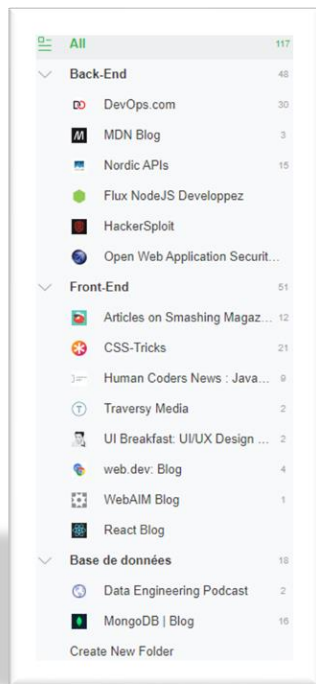
- Contrat de maintenance: mises à jour régulières, sécurité, performance,
- Planification de futures fonctionnalités comme l'intégration des tarifs, la gestion des moyens de paiements et un blog interne

Spécifications techniques

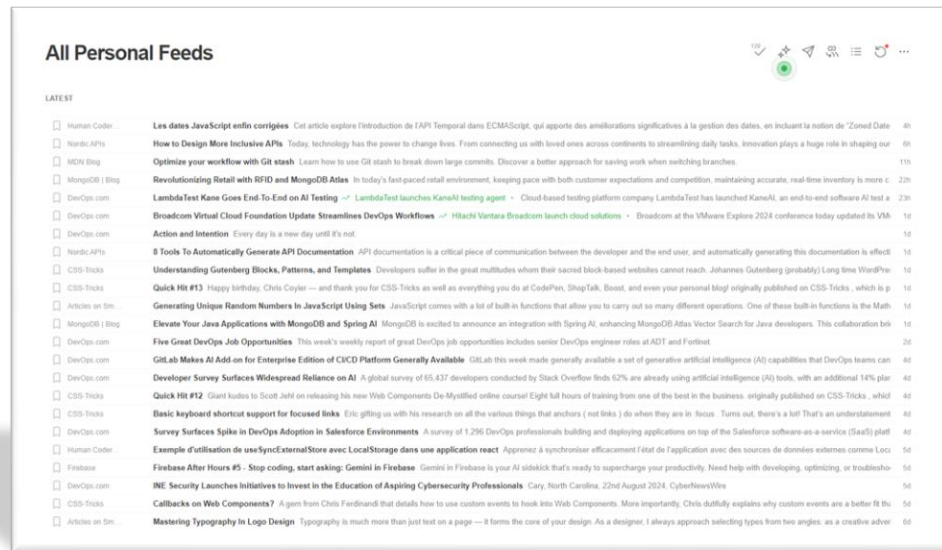


- **Description:** MongoDB, une base de données NoSQL, offre une grande flexibilité pour gérer des données non structurées grâce à son format JSON (BSON). Elle est idéale pour des applications évolutives comme Menu Maker, offrant une performance élevée, une scalabilité horizontale, et une intégration facile avec Node.js. Sa capacité à gérer des schémas dynamiques en fait un choix optimal pour stocker et gérer efficacement les données des utilisateurs et des menus.
- Ce diagramme illustre comment les données des utilisateurs sont gérées avec MongoDB. Il montre le flux des informations depuis l'interface utilisateur, à travers le backend API, jusqu'à la base de données MongoDB, en mettant en évidence la gestion des collections et les mesures de sécurité.

Veille Technologique



Sources:
 Blogs / Podcast
 Réseaux Sociaux
 Youtube
 Forums



En utilisant ces catégories de sources, la veille technologique devient plus structurée et permet de rester à jour sur les dernières avancées tout en ajustant les spécifications techniques du projet en fonction des nouvelles informations.



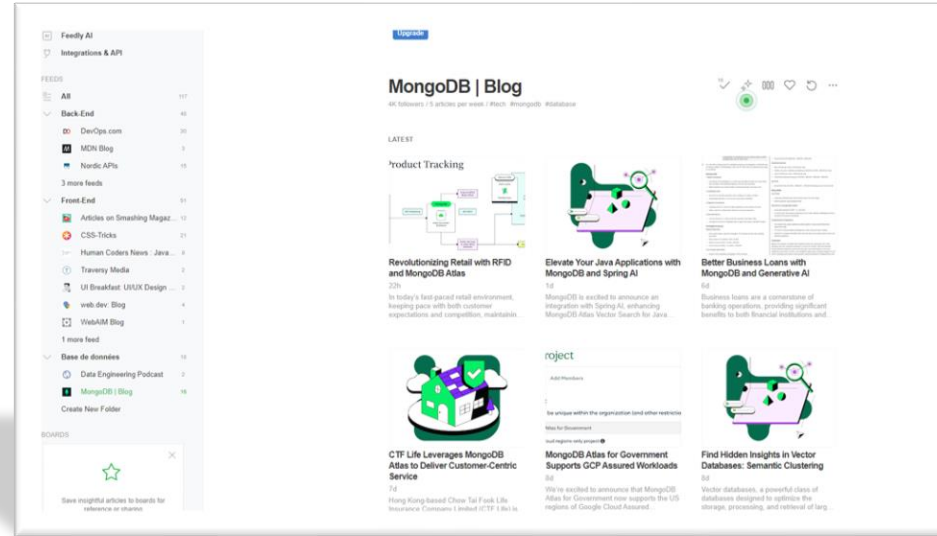
Axe 1 : Veille Spécifique - Base de Données NoSQL (MongoDB)

▪ **Objectif de Veille :**

L'objectif est de rester informé des dernières évolutions, meilleures pratiques, et cas d'utilisation de MongoDB pour s'assurer que les décisions techniques concernant la base de données du projet Menu Maker sont optimisées pour la performance, la scalabilité, et la flexibilité.

- Le blog officiel de MongoDB fournit des articles réguliers sur les nouvelles fonctionnalités, des tutoriels avancés, des études de cas de grandes entreprises, et des mises à jour sur la communauté MongoDB.

- En suivant le MongoDB Blog, l'équipe technique peut repérer des fonctionnalités importantes à intégrer dans le projet, comme l'usage des agrégations pour analyser les données des menus ou le sharding pour gérer de grandes quantités de données. Cette veille aide à faire des choix éclairés dans les spécifications techniques, garantissant une utilisation optimale de MongoDB pour le projet Menu Maker.



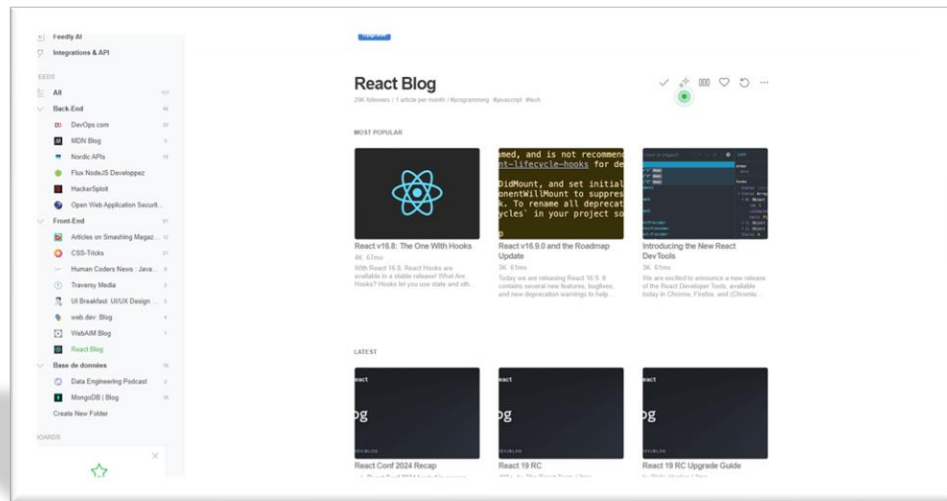
Axe 2 : Veille Spécifique - ReactJS



- **Objectif de Veille :**

L'objectif est de suivre les évolutions, meilleures pratiques, et nouvelles fonctionnalités de ReactJS pour optimiser le développement du front-end du projet Menu Maker. Cette veille permet de garantir que l'interface utilisateur est à la fois performante, dynamique, et maintenable.

- Le blog officiel de ReactJS propose des articles sur les nouvelles versions, des guides sur l'optimisation des performances, des exemples d'intégrations avancées, et des discussions sur les futures évolutions de la bibliothèque.
- Grâce à la veille sur le *React Official Blog*, l'équipe peut intégrer des pratiques optimisées dans le projet, comme l'utilisation de hooks pour gérer l'état et les effets de manière plus efficace, ou l'implémentation du code splitting pour améliorer le temps de chargement de l'application. Cette veille permet de définir des spécifications techniques qui assurent que le front-end de Menu Maker est moderne, rapide, et facile à maintenir.



Conclusion

Méthodologie Agile et Gestion de Projet :

Le projet Menu Maker adopte la méthodologie Agile, offrant flexibilité et adaptation. Scrum structure le travail en sprints pour des itérations rapides, tandis que Kanban visualise l'avancement des tâches, facilitant le suivi et les ajustements en temps réel.

Choix Technologique - MongoDB :

MongoDB est choisi pour sa gestion flexible des données non structurées et sa capacité à évoluer avec le sharding, soutenant ainsi la croissance future de l'application Menu Maker.

Veille Technologique :

Une veille technologique rigoureuse est mise en place pour suivre les évolutions de MongoDB et ReactJS. Cela permet à l'équipe de rester à jour sur les meilleures pratiques et d'intégrer les dernières innovations dans les spécifications techniques, garantissant ainsi une solution moderne, efficace et sécurisée.

Ce résumé capture les éléments clés du projet, mettant en lumière l'approche méthodologique, les choix techniques, et l'importance de la veille technologique pour assurer le succès du projet.

DES QUESTIONS ?